



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/889,350	07/17/2001	Martin Russ	36-1466	2630

7590 10/12/2005
Nixon & Vanderhye
1100 North Glebe Road 8th Floor
Arlington, VA 22201-4714

EXAMINER

BLACKWELL, JAMES H

ART UNIT PAPER NUMBER

2176

DATE MAILED: 10/12/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/889,350

Applicant(s)

RUSS ET AL.

Examiner

James H. Blackwell

Art Unit

2176

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 23 August 2005.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-17, 20, 21, 23-27 and 29-47 is/are pending in the application.
- 4a) Of the above claim(s) 18, 19, 22 and 28 is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☐ Claim(s) 1-17, 20, 21, 23-27 and 29-47 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 17 July 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☒ All b) ☐ Some * c) ☐ None of:
1. ☒ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date 10/28/04.
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____

DETAILED ACTION

1. This Office Action is in response to an after final Amendment filed **08/23/2005**.
2. Claims 1-17, 20-21, 23-27, and 29-47 remain pending in this application.
3. Claims 1, 23-24, 29, and 46-47 are independent claims.
4. Claims 29-47 are new claims.
5. Examiner notes omission of initial on previously submitted IDS (currently dated 08/23/2005) and has corrected omission.
6. Finality of the previous Office Action mailed **03/23/2005** has been withdrawn by the Examiner based on arguments by the Applicant.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

8. Claims 1-17, 20-21, 23-27, and 29-47 are rejected under 35 U.S.C. 103(a) as being unpatentable over Windows 95 (Introducing Microsoft Windows 95, copyright 1981-1995, Microsoft) in view of Wishnie et al. (hereinafter Wishnie, U.S. Patent No. 6, 148, 311, files 04/25/1997).

In regard to independent Claim 1 (and similarly independent Claims 23-24, 29, and 46-47), Windows 95 teaches *using the viewer of the operating system to view the hierarchical file structure corresponding to content for the web site* by offering as part of the operating system Windows Explorer (a viewer) where one can create, view,

and otherwise manipulate any number or type of folders, subfolders, and files within a hierarchical structure (Pg. 22, Figure; Pgs. 38, 40, Figures). Using Windows Explorer then, it would have been obvious to one of ordinary skill in the art at the time of invention to view folders, subfolders, and files comprising a web site, providing a structured and visual environment to assist in the maintenance and manipulation of folders, subfolders, and files.

Windows 95 fails to teach *and running a conversion program module to convert the file contents stored in folders arranged in the hierarchical file structure into hypermedia for the web site with hyperlinks therein corresponding to at least one relationship of at least one of the stored files with at least one other of the stored files within the hierarchical file structure*. However, Wishnie teaches a build tool that receives constructing directory information (a directory name), which identifies the directory (the root directory) containing the HTML files associated with a pre-existing web site (600). The Build tool either identifies an index.html (typically the first page loaded by a web browser when directed to a web site), or generates one. If sub-directories are located in the root directory, each is loaded (618) and the process is repeated at step 602. At the sub-directory level, an index page is created (if not located in the sub-directory) which is a parent (in the navigational structure) to all pages derived from HTML files contained in the sub-directory. The process continues until all the files and directories in the root directory have been resolved (616). Accordingly, the physical file system is manipulated to create a navigational hierarchy, which does not contain any purely structural elements. Pages with navigational structure are substituted for any structural elements

in the physical file system (Col. 7, lines 4-52). Thus, the Build tool has the ability to traverse an existing file hierarchy (which one of ordinary skill in the art at the time of invention would have typically constructed manually) containing content and process that content.

Wishnie continues by teaching that the Build routine (306) includes methods for creating HTML (*converting file contents stored in folders arranged in a hierarchical file structure into hypermedia for the web site*), creating a site file and resolving embedded content. Build routine (306) converts all of the elements associated with a single page into a standard flow based description language (HTML). Note, some of the elements of a page may already be in HTML format. For example pages imported into the web site for placement into the body (unique) section of a master page are typically written in HTML (one of ordinary skill in the art at the time of invention would understand that the master page is a *template page*). The Build routine resolves embedded content in the common portion of the master pages (the master pages can exist at each directory/sub-directory level within the directory hierarchy of files). Embedded content includes links to other web sites and navigational links (typically hyperlinks) to other pages *within the given web site*. Build routine (306) resolves the embedded content by determining the actual address or file location for the embedded content and includes this information in the HTML for a given page (Col. 7, lines 54-67; Col. 8, lines 1-16).

Wishnie continues by teaching that multimedia elements (708) include text, image data, sound, video and applets (including java applets). These types of elements are common in web pages for identifying the particular web site. For example, a master

Art Unit: 2176

page may include a company name or logo, which is to be displayed on all pages of the web site (Col. 9, lines 27-32). Thus, the Build routine can prepare all these types of content for use in a web environment. Navigational links (712) include external links, absolute links and relative links. External links provide links to locations outside the current web site. Absolute links provide a link within a web site to a fixed location in the site. Relative links are conditioned on the navigational structure of the web site and the actual addressing for such is not determined until build time. Relative links include previous, next, parent and child links. Previous and next links are links to adjacent pages at the same hierarchical level. Parent and child links are links to pages either one hierarchical level above or below a given page. For each relative link, build routine (306) *(FIG. 3) looks at the navigational structure defined for the web site and, based on the relationships of the various pages, calculates links to appropriate elements in the structure automatically at build time* (Col. 9, lines 40-56). Thus, the Build routine processes and creates *hyperlinks therein corresponding to at least one relationship of at least one of the stored files with at least one other of the stored files within the hierarchical file structure*). It is noted that though a user interface exists with the application taught by Wishnie, that there are scenarios where the interactive (or editing) need not be done within the application (to include any hierarchy, hyperlink, or file manipulations). It would have therefore been obvious to one of ordinary skill in the art at the time of invention to conclude that the application described by Wishnie would have allowed the user to process a given pre-defined directory structure containing content into a web site). It further would have been obvious to one of ordinary skill in the art at

Art Unit: 2176

the time of invention to combine the teachings of Windows 95 and Wishnie as Windows 95 provides a native File Manager to assist the user in constructing a hierarchical directory structure containing files to be converted by the software tool of Wishnie into a web site that can be uploaded to a web server, providing a convenient combination for the novice to produce a web site.

In regard to dependent Claim 2 (and similarly dependent Claim 30),

Windows 95 teaches *using the viewer to transfer files of content for the web site from other file locations of the computer configuration, into the hierarchical file structure for the web site* (Pg. 38, files can be moved from one folder to the other by dragging and dropping, see Figs).

In regard to dependent Claim 3 (and similarly dependent Claim 31),

Windows 95 fails to teach *the method including using the conversion program module to convert the files of the different formats into a form suitable for use as hypermedia on the web site*. However, Wishnie teaches that multimedia elements (708) including text, image data, sound, video and applets (including java applets) can be processed and made web compatible (Col. 9, lines 27-32). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as Windows 95 provides a native File Manager to assist the user in constructing a hierarchical directory structure containing files to be converted by the software tool of Wishnie into a web site that can be uploaded to a web server, providing a convenient combination for the novice to produce a web site.

In regard to dependent Claim 4 (and similarly dependent Claim 32),

Windows 95 fails to teach that *the conversion program module produces web pages corresponding to the first of the folders and any sub-folders with hyperlinks between them corresponding to folder hierarchy*. However, Wishnie teaches that the Build routine produces web pages derived from the content and directives found in the master pages (Col. 7, lines 54-64). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as Windows 95 provides a native File Manager to assist the user in constructing a hierarchical directory structure containing files to be converted by the software tool of Wishnie into a web site that can be uploaded to a web server, providing a convenient combination for the novice to produce a web site.

In regard to dependent Claims 5-6 (and similarly dependent Claims 33-34),

Windows 95 fails to teach that *the first folder contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the first folder, and the conversion program module produces a web page corresponding to a template for the node*. However, Wishnie teaches that at the time for build and conversion to HTML, build routine (306) resolves embedded content in the common portion of a page for all pages in the site having master pages (templates). Embedded content includes links to other web sites and navigational links to other pages within the given web site. Build routine (306) resolves the embedded content by determining the actual address or file location for the embedded content and includes this information in the HTML for a given page (Col. 8, lines 8-16). It would have been

Art Unit: 2176

obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as Windows 95 provides a native File Manager to assist the user in constructing a hierarchical directory structure containing files to be converted by the software tool of Wishnie, using templates, into a web site that can be uploaded to a web server, providing a convenient combination for the novice to produce a web site.

In regard to dependent Claim 7 (and similarly dependent Claim 35),
Windows 95 fails to teach that *the conversion program module searches the at least one sub-folder to determine if it contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the at least one sub-folder and the produces a web page corresponding to the template for a node corresponding to the at least one sub-folder*. However, Wishnie teaches that an inference engine (322) within the web site build tool receives directory information (a directory name), which identifies the directory (the root directory) containing the HTML files associated with the old web site (600). The inference engine scans the directory searching for a file named "INDEX.HTML" (602), which is typically the home page of a web site (Col. 7, lines 4-13).

Wishnie continues by teaching that the Build routine scans the entire directory hierarchy for files from which to build web pages based on master page templates (Col. 7, lines 54-64). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as Windows 95 provides a native File Manager to assist the user in constructing a hierarchical directory

structure containing files to be converted by the software tool of Wishnie by drilling down the hierarchical structure to create a web site that can be uploaded to a web server, providing a convenient combination for the novice to produce a web site.

In regard to dependent Claim 8 (and similarly dependent Claim 36),

Windows 95 fails to teach that *if no template file is found in the at least one sub-folder, the conversion program module searches said first folder to find a template file to be applied to the at least one sub-folder*. However, Wishnie teaches such a limitation (Col. 7, lines 14-52). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as Windows 95 provides a native File Manager to assist the user in constructing a hierarchical directory structure containing files to be converted by the software tool of Wishnie by drilling down the hierarchical structure to create a web site that can be uploaded to a web server, providing a convenient combination for the novice to produce a web site.

In regard to dependent Claim 9 (and similarly dependent Claim 37),

Windows 95 fails to teach that *the template file includes a plugin for inserting predetermined hypermedia from different files into the web page produced by the template*. However, Wishnie teaches such a limitation (Col. 9, lines 5-56; Fig. 7; common regions and unique regions where the main content of other pages is placed). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as Windows 95 provides a native File Manager to assist the user in constructing a hierarchical directory structure containing files to be converted by the software tool of Wishnie. Adding the teaching of

Wishnie provides the benefit of using templates to improve the convenience and accuracy of manipulating file structures and their content.

In regard to dependent Claim 10 (and similarly dependent Claim 38),

Windows 95 fails to teach that *the plugin defines a link and the conversion program module produces a hyperlink in the web page produced by means of the template with a configuration defined by the link*. However, Wishnie teaches such a limitation (Col. 9, lines 46-56; Fig. 7; master pages (templates) contain relative link conditions that are resolved based on the hierarchical structure of the site at Build time). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as Windows 95 provides a native File Manager to assist the user in constructing a hierarchical directory structure containing files to be converted by the software tool of Wishnie by converting content in combination with templates to create a web site that can be uploaded to a web server, providing a convenient combination for the novice to produce a web site.

In regard to dependent Claim 11 (and similarly dependent Claim 39),

Windows 95 fails to teach that *the first folder or the at least one sub-folder contains a document template for defining a predetermined configuration for hypermedia at a web page in the web site corresponding to a text document in the first folder or the at least one sub-folder*. However, Wishnie teaches such a limitation (Col. 9, lines 15-25; Fig. 7; master pages are tied to one or more pages in the web site). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as both inventions relate to the manipulation of

file structures for the purpose of maintaining the content contained within the file structures. Adding the teaching of Wishnie provides the benefit of using templates to assist in the creation of web sites by novices.

In regard to dependent Claim 12 (and similarly dependent Claim 40), though Windows 95 file viewer could have been used to *transfer a selected one or more of the templates from the library to the one or more of the folders*, it does not teach *providing a library of said templates*. However, the teachings of Wishnie in the use of master pages (Cols. 9-10; Col. 11, lines 1-3), lines would have suggested that it would have been obvious to one of ordinary skill in the art at the time of invention that a library of templates could have been constructed using the combination of their invention and the native file manager of Windows 95 as one produced a number of different web sites with numerous pages and corresponding master pages (templates). The user of Wishnie's invention could have then taken numerous master pages and reused them for future web page designs providing the benefit of reuse.

In regard to dependent Claim 13 (and similarly dependent Claim 41), though Windows 95 could have been used to create and visualize the claimed components and actions, it fails to explicitly teach that *the folder hierarchy includes a root folder and sub-folders depending therefrom, and including placing at least one of the templates in the root folder*. However, Wishnie teaches such a limitation (Col. 7, lines 4-17). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as both inventions relate to the manipulation of file structures for the purpose of maintaining the content contained within the file

structures. Adding the teaching of Wishnie provides the benefit of using templates to assist in the creation of web sites by novices.

In regard to dependent Claim 14 (and similarly dependent Claim 42), though Windows 95 could have been configured to access a network, and could have also manipulated the file structures using the file viewer, it fails to explicitly teach that *the computer configuration comprises a network and including arranging the files in the hierarchical file structure for the web site, using the file viewer, from different file locations in the network*. However, Wishnie teaches such a limitation (see Col. 11, lines 53-67). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as both inventions relate to the manipulation of file structures for the purpose of maintaining the content contained within the file structures. Adding the teaching of Wishnie provides the benefit of a networked computer to assist in the creation of web sites by novices.

In regard to dependent Claim 15 (and similarly dependent Claim 43), Windows 95 fails to explicitly teach *uploading the hypermedia for installation on a server for the web site*. However, Wishnie teaches that the pages are integrated and transformed into a suitable format for display (the "build") (104). Building the site includes creating a site file, creating HTML (hypertext markup language) and resolving any embedded content. Thereafter the completed site is uploaded to a web server so that it may be accessed by individuals on the World Wide Web (106) (Col. 4, lines 1-7). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as both inventions relate to the

manipulation of file structures for the purpose of maintaining the content contained within the file structures. Adding the teaching of Wishnie provides the benefit of allowing the novice to place their website where others may readily access it.

In regard to dependent Claim 16 (and similarly dependent Claim 44),
Windows 95 fails to teach *providing a local preview of the hypermedia before uploading it to the server*. However, Wishnie offers the user a view of the structure of the web site as well as views of rendered content (See Figs. 4a-b, 7). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as both inventions relate to the manipulation of file structures for the purpose of maintaining the content contained within the file structures. Adding the teaching of Wishnie provides the benefit of using a preview mode to assist in the creation of web sites by novices.

In regard to dependent Claim 17 (and similarly dependent Claim 45) and 21,
Windows 95 fails to teach *causing the conversion program module to be downloaded to the computer configuration from a remote server*. However, Wishnie teaches that the software is typically executed on a client machine, and the resulting product (web site) is uploaded to a server (see Fig. 1). It is also well known in the art of ISPs, which typically offer web storage space to customers, for them to also provide software tools for their customers. It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Windows 95 and Wishnie as both inventions relate to the manipulation of file structures for the purpose of maintaining the

Art Unit: 2176

content contained within the file structures. Adding the teaching of Wishnie provides the benefit of uniformity in customer web site design throughout the ISP.

In regard to dependent Claim 20, Windows 95 fails to explicitly teach a *program module according to claim 17 recorded on a recording medium insertable into the computer configuration to be loaded therein*. However Wishnie teaches an environment under which the application they describe can be stored (See Fig. 9), therefore making it obvious to one of ordinary skill in the art at the time of invention to combine with the native file management system provided by Windows 95, providing the benefit of having an executable program available when needed.

In regard to dependent Claim 25-27, Windows 95 teaches the operating system is executed in a networked environment for concurrent access by multiple users (Pgs. 44-58).

Response to Arguments

9. Applicant's arguments with respect to claims 1-17, 20-21, and 23-27 have been considered but are moot in view of the new ground(s) of rejection. The Applicant mainly argues that the combination of Arora, Guzak, and Hanson fails to teach or suggest files, which are converted into web pages being stored in an arrangement in a hierarchical manner using the native file system of an operating system. In light of this, the Examiner applies the teaching of Windows 95, whose operating system includes the creation, storage, manipulation, and deletion of files in combination with the teachings of Wishnie, which teaches software that operates in the operative system in concert with the native file system to perform the remainder of the claimed limitations.

Conclusion

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to James H. Blackwell whose telephone number is 571-272-4089. The examiner can normally be reached on Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Heather R. Herndon can be reached on 571-272-4136. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

11. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published

Art Unit: 2176

applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

James H. Blackwell
10/06/05

William L. Bashore
WILLIAM BASHORE
PRIMARY EXAMINER
10/7/2005